



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/995,041	11/26/2001	Konstantin I. Boudnik	SUNMP032	3339

25920 7590 08/03/2005

MARTINE PENILLA & GENCARELLA, LLP  
710 LAKEWAY DRIVE  
SUITE 200  
SUNNYVALE, CA 94085

EXAMINER

CHOW, CHIH CHING

ART UNIT PAPER NUMBER

2192

DATE MAILED: 08/03/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	<b>Application No.</b>	<b>Applicant(s)</b>	
	09/995,041	BOUDNIK ET AL.	
	<b>Examiner</b>	<b>Art Unit</b>	
	Chih-Ching Chow	2192	

**-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --**  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

#### Status

- 1) ☐ Responsive to communication(s) filed on 6/3/05.
- 2a) ☒ This action is **FINAL**.                      2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

#### Disposition of Claims

- 4) ☒ Claim(s) 1,7-9,15 and 17-20 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1,7-9,15 and 17-20 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

#### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 03 April 2002 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

#### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \*    c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

#### Attachment(s)

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)  | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                                   | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____  |

### **ETAILED ACTION**

1. This action is responsive to amendment dated June 3rd, 2005.
2. Per Applicants' request, the Specification, Claims 1, 7-9, 15, 17-20 have been amended, claims 2-6, 10-14 and 16 are canceled.
3. Claims 1, 7-9, 15, 17-20 are remain pending.

### ***Response to Amendment***

4. Applicants' amendment dated 06/03/2005, responding to the 02/28/2005 Office action provided in the objection of Specification. The examiner has reviewed the updated Specification respectfully.
5. The objection to the specification is hereby withdrawn in view of Applicants' amendment to the Specification.
6. The 35 USC 112 Rejections to Claims 6 and 14 is withdrawn since Claim 6 and 14 are canceled.

### ***Response to Arguments***

7. Applicant's arguments with respect to claims 1, 7-9, 15, 17-20 have been considered but are moot in view of the new ground(s) of rejection necessitated by Applicant's amendments to the claims. For the Applicants' convenience the amended claims and the 35 USC § 103 are listed as following.

### ***Claim Rejections - 35 USC § 103***

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

9. Claims 1, 7-9, 15, 17-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,105,148 by Pi-Yu Chung et al. (hereinafter "Chung"), in view of U.S. Patent No. 6,256,773 by Michael Bowman-Amuah (hereinafter "Bowma-Amuah").

#### CLAIM

1. A system for restoring execution of a test of a software program after interruption of the test performed in a distributed processing framework by a test application program, the system comprising:

a. a test suite configured to define tests for testing each particular software program to be tested, each test suite being configured to define a plurality of test execution requests for testing the particular software program;

c. a system controller for locating a first software test system capable of executing the first test execution request, the system controller further locating a second software test system capable of executing the second test execution request, the respective located software test system executing a respective test execution request being configured to

#### Chung / Bowan-Amuah

Chung teaches a prior art which would restore execution of an application program after interruption in a distributed network. For item a, Chung's disclosure is also used for a testing environment, see Chung's column 1 lines 44-55, "While software testing and debugging tools, such as the Purify.TM. system, provide an effective basis for detecting many programming errors which may lead to a fault in the user application process, no amount of **verification, validation or testing during the software debugging process** will detect and eliminate all software faults and give complete confidence in a user application program. Accordingly, residual faults due to untested boundary conditions, unanticipated exceptions and unexpected execution environments have been observed to escape the testing and debugging process and, when triggered during program execution, will manifest themselves and cause the application process to crash or hang, thereby causing service interruption". For items c-e, see Chung column 1, last sentence to top of column 2, "Generally, **checkpoint and restoration techniques periodically** save the process state during normal execution, and thereafter restore the saved state following a failure." Here the checkpoint data is the '**post mortem object**', checkpoint data is periodically

monitor progress in the execution of the respective test execution request on the respective located software test systems the monitoring comprising gathering data identifying each software test system that is executing at least one of the test execution requests and periodically determining a point of the execution of the respective test execution request;

d. a respective one of the located software test systems being capable of utilizing the identification of the software test system that is executing the at least one of the test execution requests of the gathered data to identify a point of the execution of a respective test execution request being executed by that one located software test system so that, upon interruption of the execution of that at least one test execution request by that respective one located software test system, the point of execution of that at least one test execution request by the one software test system is known;

e. that respective one located software test system being effective upon such interruption to use the gathered data corresponding to that interrupted execution request to reinitialize the test application program corresponding to the respective one located software test system to begin execution of the respective interrupted test execution request from a position described by the corresponding point of execution information.

updated while the application program is in operation. FIG. 1, item 70 and 75, a 'Network Interface' for a **distributed system** (*distributed processing framework*), also item 20, *Processing Node*, is an **agent process** executing on a processing resource. In Chung's disclosure, the application program can be '**reinitialized**' (restarted) from the last checkpoint, see Chung column 6, lines 42-44, "a **restart** subsystem 90 will attempt to recover the faulty application process by **initiating a restart** of the faulty application process, at its **latest checkpoint**". Also see Chung column 9, lines 31-33, "due to a failure, or the user application process has **requested** that the **checkpoint file** (*execution information*) should be stored for subsequent restoration." Further, in column 9, lines 34-49, Chung continues the description of the process in response to receiving the information request as described in claim 2, "If it is determined during step 540 that a valid checkpoint file does exist for the associated user application process, then the pre-execution checkpoint subroutine 152 will preferably return and execution of the restoration subroutine 158, discussed below in conjunction with FIGS. 8A and 8B, will preferably commence during step 550, in order to restore the data associated with the existing checkpoint file and commence execution of the user application process at the point of the restored checkpoint. ...If, however, it is determined during step 540 that a valid checkpoint file does not exist for the associated user application process, then the pre-execution checkpoint subroutine 152 will

b. a plurality of software test systems configured in a distributed processing framework, each of the test systems being configured to operate under control of a test application program to test a particular software program with respect to ability to perform specific operations, the test being configured to execute one of the test suites that defines the tests to be performed on the particular software program, first and second test execution requests for testing the particular software program being configured to be executed on different ones of the software test systems distributed among the distributed processing framework;

Claims 2-6 Canceled.

7. A system as recited in claim 1, wherein the monitoring by the system controller further includes an indication of a particular the individual test executed by the software test system executing

preferably return and execution of the user application process is preferably initiated during step 560". See Chung's disclosure, the most current checkpoint state will be restored; see Chung column 10, lines 3-7, "it can be subsequently accessed by the file system call interception subroutine 156, which implements checkpoints of the persistent state, in order to associate the persistent state checkpoints with the appropriate **(current)** volatile checkpoint".

For item b, Chung teaches all the respects of claim 1 except the test specific operations of a test suites, however Bowman teaches it in an analogous art, see Bowman column 83, lines 1-5, "the Version Control tool must be able to operate on all the platforms in use, whilst **at the same time performing Version Control for all components (*different operations, individual test pieces*) across the entire system**".

It would have been obvious to a person of ordinary skill in the art at the time of the invention was made to supplement Chung's disclosure of the checkpointing software by dividing up the software to individual components taught by Bowman-Amuah, for the purpose of viewing a computer program as a collection of largely autonomous components (Bowman-Amuah, column 4, lines 4-5).

For the feature of claim 1 see claim 1 rejection. Chung's disclosure has a way to figure out at a certain point a crash has occurred, see column 6, lines 31-39, "In a passive monitoring arrangement, each application process includes a

the one test execution request.

8. A system as recited in claim 7, wherein that respective one located software test application is effective to reinitialize the software test application program to begin execution from that particular individual test.

9. A method for restoring execution of a test of a software program after interruption of the test performed in a distributed processing framework by a software test application program, comprising:

- providing a plurality of software test systems in a distributed processing framework, the software test systems being configured to test a particular software program with respect to ability to perform specific operations, the software test application program being configured to execute a test suite that defines the tests to be performed on a particular software program;

- dividing the test suite for one of the

function from the library 150, which, when invoked by a user application process, such as the process 40, will send a heartbeat message at specified intervals to the watchdog 80, indicating that the associated process 40 is still active. If the watchdog 80 does not receive another signal (*indication*) from the application process 40 before the end of the specified interval, the watchdog 80 will presume that the application process is hung or has crashed."

For the feature of claim 7 see claim 7 rejection. Chung's disclosure allows the application to be restarted at a particular checkpoint, see column 6, lines 42-44, "a **restart** subsystem 90 will attempt to **recover** the faulty application process by initiating a **restart** (*begin execution*) of the faulty application process, at its latest checkpoint (*post mortem*)".

Chung's disclosure does restoring execution after interruption in a distributed processing framework. See Chung's Abstract, "By checkpointing and restoring a user application process, recover (*reinitializing*) of an application process from the checkpoint position is possible." For the rest of claim 9 feature see claim 1 and 8 rejections.

particular software programs to be tested into a plurality of test execution requests, first and second test execution requests for testing one particular software program to be executed on different ones of the software test systems distributed among the distributed processing framework;

monitoring progress in executing each test execution request on the respective software test system, the monitoring comprising gathering data identifying each software test system that is executing at least one of the test execution requests and periodically determining a point of the execution of the respective test execution request;

upon interruption of the execution of one of the test execution requests, using the gathered data to identify the software test system that was executing the interrupted test execution request; and

reinitializing the identified software test system that was executing the interrupted test execution request, the reinitializing being at the determined point of the execution of the respective test execution request that corresponds to the identified software test system.



Claims 10-14 canceled.

15. A method as recited in claim 9, wherein the monitoring further comprises gathering data identifying an individual test corresponding to one of the test execution requests corresponding to the identified software test system and to the point of the execution of that test execution request.

For the feature of claim 9 see claim 9 rejection, for the rest of the claim 15 feature see claim 1 and 8 rejections.

Claim 16 canceled.

17. A computer program embodied on a computer readable medium for restoring execution of testing of a software program after interruption of the testing, the testing being performed by various ones of a plurality of software test systems, each of the software test systems being configured to perform one or more tests of a test suite of the testing and being arranged in a distributed processing framework to execute a software test application program, the computer program comprising:

- a code segment that receives execution information from the software test application program, wherein the execution information includes an identification of a respective software test application program being executed, an identification of the one software test system that is executing the respective software test application program with respect to one of the tests, and a current point of execution within the test suite;

- a code segment that updates a post mortem object based on the execution information, the update being specific to the respective software test application program being executed by the one software test system that is executing the

Chung's disclosure definitely contains code segments of receiving execution information, updating checkpoint data, and reinitializing the application program, an example is shown in Chung's FIG. 10. For the rest of claim 17 feature see claim 1 and 9 rejections.

one of the tests of the test suite, and being specific to the a current point of execution within the test suite; and a code segment that reinitializes the that respective software test application program on the respective one software test system utilizing the updated post mortem object after interruption of the one test executed by the one software test system that is executing the respective software test application program.

18. A computer program as recited in claim 17, wherein the software test application program is a test harness that executes a plurality of individual tests on the software program, and wherein the point of execution refers to an individual test most recently executed by the test harness on the one software test system.

For the feature of claim 17 see claim 17 rejection, for the rest of the claim 18 feature see claim 1 rejection.

19. A computer program as recited in claim 18, wherein the update code segment updates the post mortem object with an indication of the individual test referred to by the point of execution.

For the feature of claim 18 see claim 18 rejection, for the rest of the claim 19 feature see claim 1 rejection.

20. A computer program as recited in claim 19, wherein the respective software test application program being executed by the one software test system is configured to execute a particular test corresponding to a capability of the one software test system, and wherein the reinitializing code segment reinitializes that respective software test application to execute on that one software test system to begin execution from that particular individual test specified in the corresponding updated post mortem object.

For the feature of claim 19 see claim 19 rejection, for the rest of the claim 20 feature see claim 1 and 8 rejections.

### ***Conclusion***

The following summarizes the status of the claims:

35 USC § 103 rejection: Claims 1, 7-9, 15, 17-20

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chih-Ching Chow whose telephone number is 571-272-3693. The examiner can normally be reached on 7:30am - 4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on 571-272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300. Any inquiry of a general nature of relating to the status of this application should be directed to the **TC2100 Group receptionist: 571-272-2100**.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR

Art Unit: 2192

only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>.

Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Chih-Ching Chow

Examiner

Art Unit 2192

July 29, 2005

CC

A handwritten signature in cursive script, reading "Anthony Nguyen-Ba". The signature is written in black ink and is positioned above the printed name and title.

ANTHONY NGUYEN-BA  
PRIMARY EXAMINER